IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Joahn Eker	§	Group Art Unit:	2166
Application No	10/595,984	§	Examiner:	Bruce A. Witzenburg
Filed:	02/15/2007	§	Confirmation No:	1283

Attorney Docket No: P18656-US2
Customer No.: 27045

For: UPDATING DATA IN A MOBILE TERMINAL

CERTIFICATE OF TRANSMISSION BY EFS-WEB

Date of Transmission: August 25, 2011

I hereby certify that this paper or fee is being transmitted to the United States Patent and Trademark Office electronically via EFS-Web.

Type or Print Name: Jennifer Hardin

/Jennifer Hardin/

Via EFS-Web

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313.1450

APPEAL UNDER 35 U.S.C. §134

This Brief is submitted in connection with the decision of the Primary Examiner set forth in Final Official Action dated February 4, 2011, finally rejecting claims 22-42, which are all of the pending claims in this application.

The Commissioner is hereby authorized to charge any appropriate fees under 37 C.F.R. §41.20(b)(2) that may be required by this paper, and to credit any overpayment, to Deposit Account No. 50-1379.

Real Party in Interest

The real party in interest, by assignment, is: Telefonaktiebolaget LM Ericsson (publ)
SE-164 83
Stockholm, Sweden

Related Appeals and Interferences

None.

Status of Claims

Claims 22-42 are pending in the present application, each of which are finally rejected and form the basis for this Appeal. Claims 22-42 stand rejected, under 35 U.S.C. §103(a), as being unpatentable over Levi, *et al.* (U.S. Patent No. 6,804,778) in view of Gary, *et al.* (U.S. Patent No. 5,699,509). Claims 22-42, including all amendments to the claims, are attached in the Claims Appendix. The rejection of claims 22-42 is appealed.

Status of Amendments

The claims set out in the Claims Appendix include all entered amendments. No amendment has been filed subsequent to the final rejection.

Summary of Claimed Subject Matter

Claim 22	**Specification Reference**
A method of differentially updating an image of stored data in a mobile terminal from a first data version to an updated data version, the method comprising the steps of:	Throughout the specification, including: page 3, lines 10-15
detecting whether the image of stored data in a flash memory of the mobile terminal includes one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version;	Throughout the specification, including: page 3, lines 18-22
receiving dedicated differential update instructions, wherein;	Throughout the specification, including: page 8, lines 16-18
the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair	Throughout the specification, including: page 9, lines 20-31

the data that is inconsistent with the first data version, and	
the dedicated differential update instructions are generated in response to detecting the image of stored data in the flash memory of the mobile terminal includes one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and	Throughout the specification, including: page 22, lines 4-12
loading the dedicated differential update instructions into the flash memory of the mobile terminal;	Throughout the specification, including page 22, lines 15-19
repairing, when generating the updated data version, any such detected corrupted memory block; wherein the image of stored data in the flash memory is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.	Throughout the specification, including: page 9, lines 7-12

Claim 23	**Specification Reference**
The method according to claim 22, further comprising generating the differential update instructions based on information about detected corrupted memory blocks, if any.	Throughout the specification, including: page 22, lines 4-12

Claim 37	**Specification Reference**
A mobile terminal comprising:	Throughout the specification, including: page 8, line 14
a flash memory for storing an image of data;	Throughout the specification, including: page 8, line 15
communications means adapted to receive from a data processing system dedicated differential update instructions for updating the image of data stored in the flash memory from a first data version to an updated data version;	Throughout the specification, including: page 8, lines 16-18
processing means adapted to generate the updated data version from the image of the stored data and from the received dedicated	Throughout the specification, including: page 8, lines 19-20

differential update instructions, wherein the processing means is further adapted to:	
generate information from the image of the stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version;	Throughout the specification, including: page 9, lines 20-31
communicate the generated information via the communications means to the data processing system for generating the dedicated differential update instructions, wherein	Throughout the specification, including: page 8, lines 24-26
the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and	Throughout the specification, including: page 9, lines 20-31
the dedicated differential update instructions are generated in response to the data processing system receiving the information from the image of stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and	Throughout the specification, including: page 22, lines 4-12
repair any such detected corrupted memory block; wherein the image of stored data in the flash memory is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.	Throughout the specification, including: page 9, lines 7-12

Claim 38	**Specification Reference**
A data processing system for facilitating differentially updating an image of stored data in a mobile terminal from a first data version to an updated data version, the data processing system comprising:	Throughout the specification, including: page 8, line 28 – page 9, line 2
means for loading dedicated differential update instructions into a flash memory of the mobile terminal, the dedicated differential update instructions causing a processor of the mobile terminal to generate the updated data version from the an image of stored data and	Throughout the specification, including: page 8, line 30 – page 9, line 2.

the loaded dedicated differential update instructions;	
the data processing system further comprising:	Throughout the specification, including: page 8, line 28 – page 9, line 2
means for receiving information from the mobile terminal indicative of the presence of one or more corrupted memory blocks wherein the image of stored data is inconsistent with the first data version; and	Throughout the specification, including: page 9, lines 4-6
processing means adapted to generate the dedicated differential update instructions from the first and updated data versions and from received information, wherein	Throughout the specification, including: page 9, lines 7-9
the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and	Throughout the specification, including: page 9, lines 20-31
the dedicated differential update instructions are generated in response to receiving information from the mobile terminal indicating that the image of stored data in the flash memory includes one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and	Throughout the specification, including: page 22, lines 4-12
wherein the differential update instructions used to repair the data that is inconsistent with the first data version are adapted to cause the processor of the mobile terminal to repair any such detected corrupted memory block; wherein the image of stored data in the flash memory of the mobile terminal is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.	Throughout the specification, including: page 9, lines 7-12

Claim 39	**Specification Reference**
A computer program comprising program code means embodied on a computer readable medium to be loaded into a flash memory means and executed by a processor means and adapted to cause a mobile terminal to differentially update an image of stored data	Throughout the specification, including: page 9, lines 14-18

in the flash memory of the mobile terminal from a first data version to an updated data version by performing the following steps, when the computer program is executed by the processor means on the mobile terminal:	
generating information from the image of stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with a first data version;	Throughout the specification, including: page 9, lines 19-23
loading dedicated differential update instructions into the processor of the mobile terminal, wherein	Throughout the specification, including: page 9, lines 27-28
the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and	Throughout the specification, including: page 9, lines 20-31
the dedicated differential update instructions are generated in response to the information from the image of stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and	Throughout the specification, including: page 22, lines 4-12
generating the updated data version by the processor of the mobile terminal from the stored data and the loaded dedicated differential update instructions, including repairing any such detected corrupted memory block; wherein the image of stored data in the flash memory of the mobile terminal is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.	Throughout the specification, including: page 9, lines 7-12

Claim 40	**Specification Reference**
A computer program comprising program code means embodied on a computer readable medium to be loaded into a memory means and executed by a processor means and adapted to cause a data processing system to facilitate differentially updating an image of stored data in a flash memory of a	Throughout the specification, including: page 9, lines 14-18

mobile terminal from a first data version to an updated data version by performing the following steps, when the computer program is executed by the processor on the data processing system:	
generating dedicated differential update instructions from the first and updated data versions and from information received from the mobile terminal, wherein the received information is indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version, wherein	Throughout the specification, including: page 9, lines 7-12
the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and	Throughout the specification, including: page 22, lines 4-12
the dedicated differential update instructions are generated in response to receiving the received information; and	Throughout the specification, including: page 22, lines 4-12
loading the generated dedicated differential update instructions into the flash memory of the mobile terminal, the dedicated differential update instructions causing the processor of the mobile terminal to generate the updated data version from the stored data and the loaded dedicated differential update instructions, wherein the image of stored data in the flash memory of the mobile terminal is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.	Throughout the specification, including: page 9, lines 27-28

The specification references listed above are provided solely to comply with the USPTO's current regulations regarding appeal briefs. The use of such references should not be interpreted to limit the scope of the claims to such references, nor to limit the scope of the claimed invention in any manner.

Grounds of Rejection to be Reviewed on Appeal

The rejection of claims 22-42 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,804,778 ("Levi") and U.S. Patent 5,669,509 ("Gary") is to be reviewed on appeal.

Argument

The rejection of claims 22-42 under 35 U.S.C. § 103(a) as being unpatentable over Levi and Gary should be reversed

In the Final Office Action dated February 4, 2011 ("Final Office Action"), Examiner rejected claims 22-42 under 35 U.S.C. § 103(a) as being unpatentable over Levi and Gary. While not conceding that the references qualify as prior art, but instead to expedite prosecution, Appellant has chosen to respectfully disagree and shows that the rejection in the Final Office Action is in clear error. Appellant reserves the right, for example, in a continuing application, to establish that the cited references, or other references cited now or hereafter, do not qualify as prior art as to an invention embodiment previously, currently, or subsequently claimed.

Appellant respectfully submits that claims 22-42 are patentable at least because Levi and Gary, taken alone or in any permissible combination, fail to disclose, teach, or even suggest the elements of the independent claims, as discussed herein.

A. Independent claim 22

For example, Levi and Gary, taken alone or in any permissible combination, fail to disclose, teach, or even suggest "receiving dedicated differential update instructions, wherein the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version," as recited in independent claim 22.

The aforementioned element of independent claim 22 explicitly recites that the "dedicated differential update instructions" comprise two types of "differential update instructions": (1) "differential update instructions used to generate the updated data version" and (2) "differential update instructions used to repair the data that is inconsistent with the first data version." In support of the rejection, Examiner cites various passages of Levi (col. 2, lines 38-59; col. 3, lines 43-55; col. 4, lines 24-61; and col. 12, lines 47-63) as allegedly disclosing the aforementioned claim element. Appellant will address each cited passage of Levi in turn:

Col. 2, lines 38-59 of Levi discusses:

> *An aspect of some preferred embodiments of the invention relate to data redress by an output monitor. In a preferred embodiment of the invention, a copy of some or all of the data which can be transmitted is stored at a secure location. When data is proscribed from being transmitted, for example for reason of it being tampered, the output monitor obtains a "clean" copy of the data from the secure location and transmits the clean data instead. In some cases the clean data may be more limited than the original data, for example a message which indicates that data is not being transmitted. Alternatively, proscribed data is not transmitted, so that transmitted WWW pages contain blank areas. Alternatively, a standard message is transmitted, to fill in the blank areas. Alternatively or additionally, the transmitted WWW page is modified so that the page appears not to be missing data and/or so that the distortion of the page is minimized. Alternatively, the altered data is allowed to go out, with an additional message, for example, to warn the user of possible corruption. An example of such a message is a disclaimer of warranty for the content of the data. Another example of a message is a warning that the data may be incorrect.*

In other words, this particular passage of Levi merely discusses "data redress by an output monitor." According to the passage of Levi, a substitute "clean copy" is sent, a "modified" copy is sent, or the "altered data" is sent, along with a notice that the data has been altered. However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to receiving different types of instructions, much less receiving "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 3, lines 45-55 of Levi discusses:

> *Alternatively or additionally, transmitting said data comprises not transmitting said data if said quality is not assured. Alternatively or additionally, said transmitting said data comprises transmitting said data if said data does not require quality assurance. Alternatively or additionally, the method comprises redressing said data if said quality assurance fails. Preferably, redressing comprises replacing said data with verified data.*

Preferably, said verified data is a copy of the data which was to be obtained by said data provider. Alternatively said verified data is not up-to-date copy of the data which was to be obtained by said data provider.

In other words, this passage of Levi merely discusses different ways of "transmitting data," "redressing said data," and replaying data with "verified data." However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 4, lines 24-61 of Levi discusses:

In a preferred embodiment of the invention, verifying said first stage data comprises verifying a signature on a program used for said first manufacturing step. Alternatively or additionally, said first stage signing is performed by a program which performs said first manufacturing step. Alternatively or additionally, said first manufacturing step comprises database querying. Alternatively or additionally, said first manufacturing step comprises retrieving data from a remote source. Alternatively or additionally, said first and said second manufacturing steps are performed at a computing site. Alternatively or additionally, the method comprises redressing said data if said verification fails. Preferably, said redressing comprises performing a backup data manufacturing process.

There is also provided in accordance with a preferred embodiment of the invention, a method of data corruption recovery, comprising:

detecting that data to be transmitted is corrupted, after said data is prepared for transmission and while transmitting said data;

redressing said data; and

transmitting said redressed data instead of said corrupted data. Preferably, said redressing comprises retrieving replacement data from a secured location. Preferably, said secured location contains a copy of said corrupted data. Alternatively or additionally, said secured location contains a previous version of said corrupted data. Alternatively or additionally, said secured location contains a less up-to-date copy of said corrupted data.

In a preferred embodiment of the invention, said redressing comprises retrieving replacement data from a remote location. Alternatively or additionally, said redressing comprises modifying a data transmission to not include a reference to said corrupted data. Alternatively or additionally, said redressing comprises manufacturing replacement data for said corrupted data. Alternatively or additionally, said redressing is transparent to a receiver of said data transmission.

At most, the cited passage merely describes the transmission of data to correct corrupted data. However, nothing in the cited passage discloses, teaches or even suggests anything about the transmission or receipt of different type of instructions,

much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between the transmission of data to correct corrupted data and the transmission of instructions "used to generate the updated data version" and instructions "used to repair the data that is inconsistent with the first data version."

Col. 12, lines 47-63 of Levi discusses:

> *In a preferred embodiment of the invention, the backup data is a copy of the original data, maintained at a secure location which is accessible only, or mainly, by the output monitor. Thus, there is a greater probability that the backup data is not tainted. Possibly, the backup data is encrypted or stamped with a digital signature. Possibly, a copy of all the data is maintained. In one example, only data which is susceptible to corruption is maintained in copy. In another example, only data which must be available is maintained as a copy. Possibly, if data corruption is detected, a copy of the data is used to replace the site database, possibly automatically.*

At most, the cited passage merely discusses aspects of Levi's backup data. Nothing in the cited passage discloses, teaches or even suggests anything about receiving different type of instructions, much less receiving "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Regarding claim 23, nothing in Levi and Gary, taken alone or in any permissible combination, discloses, teaches, or even suggests "generating the differential update instructions based on information about detected corrupted memory blocks, if any." As previously discussed, nothing in the cited references discusses anything about "differential update instructions," much less generating such instructions "based on information about detected corrupted memory blocks."

In the "Response to Arguments" section of the Final Office Action, Examiner asserts that "because there is no differentiating language separating the updating data version and what is effectively the repaired first version, the examine (*sic*) does not believe the amendments presented fully overcome the cited prior art." Applicant respectfully disagrees. In fact, the claims explicitly distinguish between "a first data version" and "an updated data version." *See*, preamble, claim 22 ("updating an image of stored data in a mobile terminal from a first data version to an updated data version").

Thus, something that is "effectively the repaired first version" is expressly distinguished from "an updated data version" since the "first version" (repaired or not) is updated to "an updated data version," according to the present claims.

Examiner issued an Advisory Action dated May 12, 2011 ("Advisory Action") that maintains the rejection. Examiner states that "Applicant argues Levi fails to disclose the two components of 'differential updates' one used to 'generate the updated data version' and one used to 'repair data that is inconsistent' however the claimed language does not necessarily individualize these components. Because they are not defined individually, data which repairs an inconsistent version also generates data updated data as the previous data was detected as corrupt and in need of update." Appellant respectfully submits that Examiner's rationale for maintaining the rejection is in error.

As an initial matter, Appellant respectfully submits that independent claim 22 does not recite "differential updates" as asserted by Examiner, but in fact, independent claim 22 recites "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between sending "clean" or "modified" data, as discussed in Levi and receiving "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Furthermore, even if Levi's "clean" or "modified" data could be analogized to the claimed "instructions" (a point that Appellant does not concede), Appellant disagrees with Examiner's assertion that the claimed language "does not necessarily individualize these components." Actually, the claimed language of claim 22 explicitly recites that the claimed "dedicated differential update instructions" comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version," which are clearly two types of differential update instructions.

Thus, for at least these reasons, independent claim 22 (and all claims dependent therefrom) are patentable over Levi and Gary, taken alone or in any permissible combination. Appellant respectfully requests that the rejection be reversed.

B. Independent Claim 37

For example, Levi and Gary, taken alone or in any permissible combination, fail to disclose, teach, or even suggest "wherein the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version," as recited in independent claim 37.

The aforementioned element of independent claim 37 explicitly recites that the "dedicated differential update instructions" comprise two types of "differential update instructions": (1) "differential update instructions used to generate the updated data version" and (2) "differential update instructions used to repair the data that is inconsistent with the first data version." In support of the rejection, Examiner cites various passages of Levi (col. 2, lines 38-59; col. 3, lines 43-55; col. 4, lines 24-61; and col. 12, lines 47-63) as allegedly disclosing the aforementioned claim element. Appellant will address each cited passage of Levi in turn:

Col. 2, lines 38-59 of Levi discusses:

> *An aspect of some preferred embodiments of the invention relate to data redress by an output monitor. In a preferred embodiment of the invention, a copy of some or all of the data which can be transmitted is stored at a secure location. When data is proscribed from being transmitted, for example for reason of it being tampered, the output monitor obtains a "clean" copy of the data from the secure location and transmits the clean data instead. In some cases the clean data may be more limited than the original data, for example a message which indicates that data is not being transmitted. Alternatively, proscribed data is not transmitted, so that transmitted WWW pages contain blank areas. Alternatively, a standard message is transmitted, to fill in the blank areas. Alternatively or additionally, the transmitted WWW page is modified so that the page appears not to be missing data and/or so that the distortion of the page is minimized. Alternatively, the altered data is allowed to go out, with an additional message, for example, to warn the user of possible corruption. An example of such a message is a disclaimer of warranty for the content of the data. Another example of a message is a warning that the data may be incorrect.*

In other words, this particular passage of Levi merely discusses "data redress by an output monitor" According to the passage of Levi, a substitute "clean copy" is sent, a "modified" copy is sent, or the "altered data" is sent, along with a notice that the data has been altered. However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to receiving different types of

instructions, much less receiving "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 3, lines 45-55 of Levi discusses:

> *Alternatively or additionally, transmitting said data comprises not transmitting said data if said quality is not assured. Alternatively or additionally, said transmitting said data comprises transmitting said data if said data does not require quality assurance. Alternatively or additionally, the method comprises redressing said data if said quality assurance fails. Preferably, redressing comprises replacing said data with verified data. Preferably, said verified data is a copy of the data which was to be obtained by said data provider. Alternatively said verified data is not up-to-date copy of the data which was to be obtained by said data provider.*

In other words, this passage of Levi merely discusses different ways of "transmitting data," "redressing said data," and replaying data with "verified data." However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 4, lines 24-61 of Levi discusses:

> *In a preferred embodiment of the invention, verifying said first stage data comprises verifying a signature on a program used for said first manufacturing step. Alternatively or additionally, said first stage signing is performed by a program which performs said first manufacturing step. Alternatively or additionally, said first manufacturing step comprises database querying. Alternatively or additionally, said first manufacturing step comprises retrieving data from a remote source. Alternatively or additionally, said first and said second manufacturing steps are performed at a computing site. Alternatively or additionally, the method comprises redressing said data if said verification fails. Preferably, said redressing comprises performing a backup data manufacturing process.*
>
> *There is also provided in accordance with a preferred embodiment of the invention, a method of data corruption recovery, comprising:*
>
> *detecting that data to be transmitted is corrupted, after said data is prepared for transmission and while transmitting said data;*
>
> *redressing said data; and*
>
> *transmitting said redressed data instead of said corrupted data. Preferably, said redressing comprises retrieving replacement data from a secured location. Preferably, said secured location contains a copy of said corrupted data. Alternatively or additionally, said secured location contains a previous version of said corrupted data. Alternatively or additionally, said secured location contains a less up-to-date copy of said corrupted data.*

> *In a preferred embodiment of the invention, said redressing comprises retrieving replacement data from a remote location. Alternatively or additionally, said redressing comprises modifying a data transmission to not include a reference to said corrupted data. Alternatively or additionally, said redressing comprises manufacturing replacement data for said corrupted data. Alternatively or additionally, said redressing is transparent to a receiver of said data transmission.*

At most, the cited passage merely describes the transmission of data to correct corrupted data. However, nothing in the cited passage discloses, teaches or even suggests anything about the transmission or receipt of different type of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between the transmission of data to corrected corrupted data and the receipt of instructions "used to generate the updated data version" and instructions "used to repair the data that is inconsistent with the first data version."

Col. 12, lines 47-63 of Levi discusses:

> *In a preferred embodiment of the invention, the backup data is a copy of the original data, maintained at a secure location which is accessible only, or mainly, by the output monitor. Thus, there is a greater probability that the backup data is not tainted. Possibly, the backup data is encrypted or stamped with a digital signature. Possibly, a copy of all the data is maintained. In one example, only data which is susceptible to corruption is maintained in copy. In another example, only data which must be available is maintained as a copy. Possibly, if data corruption is detected, a copy of the data is used to replace the site database, possibly automatically.*

At most, the cited passage merely discusses aspects of Levi's backup data. Nothing in the cited passage discloses, teaches or even suggests anything about different type of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

In the "Response to Arguments" section of the Final Office Action, Examiner asserts that "because there is no differentiating language separating the updating data version and what is effectively the repaired first version, the examine (*sic*) does not believe the amendments presented fully overcome the cited prior art." Applicant respectfully disagrees. In fact, the claims explicitly distinguish between "a first data

version" and "an updated data version." *See*, claim 37 ("updating an image of stored data in the flash memory from a first data version to an updated data version"). Thus, something that is "effectively the repaired first version" is expressly distinguished from "an updated data version" since the "first version" (repaired or not) is updated to "an updated data version," according to the present claims.

Examiner issued an Advisory Action dated May 12, 2011 ("Advisory Action") that maintains the rejection. Examiner states that "Applicant argues Levi fails to disclose the two components of 'differential updates' one used to 'generate the updated data version' and one used to 'repair data that is inconsistent' however the claimed language does not necessarily individualize these components. Because they are not defined individually, data which repairs an inconsistent version also generates data updated data as the previous data was detected as corrupt and in need of update." Appellant respectfully submits that Examiner's rationale for maintaining the rejection is in error.

As an initial matter, Appellant respectfully submits that independent claim 37 does not recite "differential updates" as asserted by Examiner, but in fact, independent claim 37 recites "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between sending "clean" or "modified" data, as discussed in Levi and "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Furthermore, even if Levi's "clean" or "modified" data could be analogized to the claimed "instructions" (a point that Appellant does not concede), Appellant disagrees with Examiner's assertion that the claimed language "does not necessarily individualize these components." Actually, the claimed language of claim 37 explicitly recites that the claimed "dedicated differential update instructions" comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version," which are clearly two types of differential update instructions.

Thus, for at least these reasons, independent claim 37 are patentable over Levi and Gary, taken alone or in any permissible combination. Appellant respectfully requests that the rejection be reversed.

C. Independent Claim 38

For example, Levi and Gary, taken alone or in any permissible combination, fail to disclose, teach, or even suggest "wherein the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version," as recited in independent claim 38.

The aforementioned element of independent claim 38 explicitly recites that the "dedicated differential update instructions" comprise two types of "differential update instructions": (1) "differential update instructions used to generate the updated data version" and (2) "differential update instructions used to repair the data that is inconsistent with the first data version." In support of the rejection, Examiner cites various passages of Levi (col. 2, lines 38-59; col. 3, lines 43-55; col. 4, lines 24-61; and col. 12, lines 47-63) as allegedly disclosing the aforementioned claim element. Appellant will address each cited passage of Levi in turn:

Col. 2, lines 38-59 of Levi discusses:

> *An aspect of some preferred embodiments of the invention relate to data redress by an output monitor. In a preferred embodiment of the invention, a copy of some or all of the data which can be transmitted is stored at a secure location. When data is proscribed from being transmitted, for example for reason of it being tampered, the output monitor obtains a "clean" copy of the data from the secure location and transmits the clean data instead. In some cases the clean data may be more limited than the original data, for example a message which indicates that data is not being transmitted. Alternatively, proscribed data is not transmitted, so that transmitted WWW pages contain blank areas. Alternatively, a standard message is transmitted, to fill in the blank areas. Alternatively or additionally, the transmitted WWW page is modified so that the page appears not to be missing data and/or so that the distortion of the page is minimized. Alternatively, the altered data is allowed to go out, with an additional message, for example, to warn the user of possible corruption. An example of such a message is a disclaimer of warranty for the content of the data. Another example of a message is a warning that the data may be incorrect.*

In other words, this particular passage of Levi merely discusses "data redress by an output monitor." According to the passage of Levi, a substitute "clean copy" is sent, a

"modified" copy is sent, or the "altered data" is sent, along with a notice that the data has been altered. However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to receiving different types of instructions, much less receiving "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 3, lines 45-55 of Levi discusses:

> *Alternatively or additionally, transmitting said data comprises not transmitting said data if said quality is not assured. Alternatively or additionally, said transmitting said data comprises transmitting said data if said data does not require quality assurance. Alternatively or additionally, the method comprises redressing said data if said quality assurance fails. Preferably, redressing comprises replacing said data with verified data. Preferably, said verified data is a copy of the data which was to be obtained by said data provider. Alternatively said verified data is not up-to-date copy of the data which was to be obtained by said data provider.*

In other words, this passage of Levi merely discusses different ways of "transmitting data," "redressing said data," and replaying data with "verified data." However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 4, lines 24-61 of Levi discusses:

> *In a preferred embodiment of the invention, verifying said first stage data comprises verifying a signature on a program used for said first manufacturing step. Alternatively or additionally, said first stage signing is performed by a program which performs said first manufacturing step. Alternatively or additionally, said first manufacturing step comprises database querying. Alternatively or additionally, said first manufacturing step comprises retrieving data from a remote source. Alternatively or additionally, said first and said second manufacturing steps are performed at a computing site. Alternatively or additionally, the method comprises redressing said data if said verification fails. Preferably, said redressing comprises performing a backup data manufacturing process.*
>
> *There is also provided in accordance with a preferred embodiment of the invention, a method of data corruption recovery, comprising:*
>
> *detecting that data to be transmitted is corrupted, after said data is prepared for transmission and while transmitting said data;*
>
> *redressing said data; and*

> *transmitting said redressed data instead of said corrupted data. Preferably, said redressing comprises retrieving replacement data from a secured location. Preferably, said secured location contains a copy of said corrupted data. Alternatively or additionally, said secured location contains a previous version of said corrupted data. Alternatively or additionally, said secured location contains a less up-to-date copy of said corrupted data.*
>
> *In a preferred embodiment of the invention, said redressing comprises retrieving replacement data from a remote location. Alternatively or additionally, said redressing comprises modifying a data transmission to not include a reference to said corrupted data. Alternatively or additionally, said redressing comprises manufacturing replacement data for said corrupted data. Alternatively or additionally, said redressing is transparent to a receiver of said data transmission.*

At most, the cited passage merely describes the transmission of data to correct corrupted data. However, nothing in the cited passage discloses, teaches or even suggests anything about the transmission or receipt of different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between the transmission of data to correct corrupted data and the transmission of instructions "used to generate the updated data version" and instructions "used to repair the data that is inconsistent with the first data version."

Col. 12, lines 47-63 of Levi discusses:

> *In a preferred embodiment of the invention, the backup data is a copy of the original data, maintained at a secure location which is accessible only, or mainly, by the output monitor. Thus, there is a greater probability that the backup data is not tainted. Possibly, the backup data is encrypted or stamped with a digital signature. Possibly, a copy of all the data is maintained. In one example, only data which is susceptible to corruption is maintained in copy. In another example, only data which must be available is maintained as a copy. Possibly, if data corruption is detected, a copy of the data is used to replace the site database, possibly automatically.*

At most, the cited passage merely discusses aspects of Levi's backup data. Nothing in the cited passage discloses, teaches or even suggests anything about different type of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

In the "Response to Arguments" section of the Final Office Action, Examiner asserts that "because there is no differentiating language separating the updating data

version and what is effectively the repaired first version, the examine (*sic*) does not believe the amendments presented fully overcome the cited prior art." Applicant respectfully disagrees. In fact, the claims explicitly distinguish between "a first data version" and "an updated data version." *See*, claim 38 ("updating an image of stored data in a mobile terminal from a first data version to an updated data version"). Thus, something that is "effectively the repaired first version" is expressly distinguished from "an updated data version" since the "first version" (repaired or not) is updated to "an updated data version," according to the present claims.

Examiner issued an Advisory Action dated May 12, 2011 ("Advisory Action") that maintains the rejection. Examiner states that "Applicant argues Levi fails to disclose the two components of 'differential updates' one used to 'generate the updated data version' and one used to 'repair data that is inconsistent' however the claimed language does not necessarily individualize these components. Because they are not defined individually, data which repairs an inconsistent version also generates data updated data as the previous data was detected as corrupt and in need of update." Appellant respectfully submits that Examiner's rationale for maintaining the rejection is in error.

As an initial matter, Appellant respectfully submits that independent claim 38 does not recite "differential updates" as asserted by Examiner, but in fact, independent claim 38 recites "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between sending "clean" or "modified" data, as discussed in Levi and "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Furthermore, even if Levi's "clean" or "modified" data could be analogized to the claimed "instructions" (a point that Appellant does not concede), Appellant disagrees with Examiner's assertion that the claimed language "does not necessarily individualize these components." Actually, the claimed language of claim 38 explicitly recites that the claimed "dedicated differential update instructions" comprises "differential update instructions used to generate the updated data version" and "differential update

instructions used to repair the data that is inconsistent with the first data version," which are clearly two types of differential update instructions.

Thus, for at least these reasons, independent claim 38 are patentable over Levi and Gary, taken alone or in any permissible combination. Appellant respectfully requests that the rejection be reversed.

D. Independent Claim 39

For example, Levi and Gary, taken alone or in any permissible combination, fail to disclose, teach, or even suggest "wherein the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version," as recited in independent claim 39.

The aforementioned element of independent claim 39 explicitly recites that the "dedicated differential update instructions" comprise two types of "differential update instructions": (1) "differential update instructions used to generate the updated data version" and (2) "differential update instructions used to repair the data that is inconsistent with the first data version." In support of the rejection, Examiner cites various passages of Levi (col. 2, lines 38-59; col. 3, lines 43-55; col. 4, lines 24-61; and col. 12, lines 47-63) as allegedly disclosing the aforementioned claim element. Appellant will address each cited passage of Levi in turn:

Col. 2, lines 38-59 of Levi discusses:

> *An aspect of some preferred embodiments of the invention relate to data redress by an output monitor. In a preferred embodiment of the invention, a copy of some or all of the data which can be transmitted is stored at a secure location. When data is proscribed from being transmitted, for example for reason of it being tampered, the output monitor obtains a "clean" copy of the data from the secure location and transmits the clean data instead. In some cases the clean data may be more limited than the original data, for example a message which indicates that data is not being transmitted. Alternatively, proscribed data is not transmitted, so that transmitted WWW pages contain blank areas. Alternatively, a standard message is transmitted, to fill in the blank areas. Alternatively or additionally, the transmitted WWW page is modified so that the page appears not to be missing data and/or so that the distortion of the page is minimized. Alternatively, the altered data is allowed to go out, with an additional message, for example, to warn the user of possible corruption. An example of such a message is a disclaimer of warranty for the content of the data. Another example of a message is a warning that the data may be incorrect.*

In other words, this particular passage of Levi merely discusses "data redress by an output monitor." According to the passage of Levi, a substitute "clean copy" is sent, a "modified" copy is sent, or the "altered data" is sent, along with a notice that the data has been altered. However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to receiving different types of instructions, much less receiving "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 3, lines 45-55 of Levi discusses:

> *Alternatively or additionally, transmitting said data comprises not transmitting said data if said quality is not assured. Alternatively or additionally, said transmitting said data comprises transmitting said data if said data does not require quality assurance. Alternatively or additionally, the method comprises redressing said data if said quality assurance fails. Preferably, redressing comprises replacing said data with verified data. Preferably, said verified data is a copy of the data which was to be obtained by said data provider. Alternatively said verified data is not up-to-date copy of the data which was to be obtained by said data provider.*

In other words, this passage of Levi merely discusses different ways of "transmitting data," "redressing said data," and replaying data with "verified data." However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 4, lines 24-61 of Levi discusses:

> *In a preferred embodiment of the invention, verifying said first stage data comprises verifying a signature on a program used for said first manufacturing step. Alternatively or additionally, said first stage signing is performed by a program which performs said first manufacturing step. Alternatively or additionally, said first manufacturing step comprises database querying. Alternatively or additionally, said first manufacturing step comprises retrieving data from a remote source. Alternatively or additionally, said first and said second manufacturing steps are performed at a computing site. Alternatively or additionally, the method comprises redressing said data if said verification fails. Preferably, said redressing comprises performing a backup data manufacturing process.*
>
> *There is also provided in accordance with a preferred embodiment of the invention, a method of data corruption recovery, comprising:*

detecting that data to be transmitted is corrupted, after said data is prepared for transmission and while transmitting said data;

redressing said data; and

transmitting said redressed data instead of said corrupted data. Preferably, said redressing comprises retrieving replacement data from a secured location. Preferably, said secured location contains a copy of said corrupted data. Alternatively or additionally, said secured location contains a previous version of said corrupted data. Alternatively or additionally, said secured location contains a less up-to-date copy of said corrupted data.

In a preferred embodiment of the invention, said redressing comprises retrieving replacement data from a remote location. Alternatively or additionally, said redressing comprises modifying a data transmission to not include a reference to said corrupted data. Alternatively or additionally, said redressing comprises manufacturing replacement data for said corrupted data. Alternatively or additionally, said redressing is transparent to a receiver of said data transmission.

At most, the cited passage merely describes the transmission of data to correct corrupted data. However, nothing in the cited passage discloses, teaches or even suggests anything about the transmission or receipt of different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between the transmission of data to correct corrupted data and the transmission of instructions "used to generate the updated data version" and instructions "used to repair the data that is inconsistent with the first data version."

Col. 12, lines 47-63 of Levi discusses:

In a preferred embodiment of the invention, the backup data is a copy of the original data, maintained at a secure location which is accessible only, or mainly, by the output monitor. Thus, there is a greater probability that the backup data is not tainted. Possibly, the backup data is encrypted or stamped with a digital signature. Possibly, a copy of all the data is maintained. In one example, only data which is susceptible to corruption is maintained in copy. In another example, only data which must be available is maintained as a copy. Possibly, if data corruption is detected, a copy of the data is used to replace the site database, possibly automatically.

At most, the cited passage merely discusses aspects of Levi's backup data. Nothing in the cited passage discloses, teaches or even suggests anything about different type of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

In the "Response to Arguments" section of the Final Office Action, Examiner asserts that "because there is no differentiating language separating the updating data version and what is effectively the repaired first version, the examine (*sic*) does not believe the amendments presented fully overcome the cited prior art." Applicant respectfully disagrees. In fact, the claims explicitly distinguish between "a first data version" and "an updated data version." *See*, claim 39 ("update an image of stored data in a flash memory of the mobile terminal from a first data version to an updated data version"). Thus, something that is "effectively the repaired first version" is expressly distinguished from "an updated data version" since the "first version" (repaired or not) is updated to "an updated data version," according to the present claims.

Examiner issued an Advisory Action dated May 12, 2011 ("Advisory Action") that maintains the rejection. Examiner states that "Applicant argues Levi fails to disclose the two components of 'differential updates' one used to 'generate the updated data version' and one used to 'repair data that is inconsistent' however the claimed language does not necessarily individualize these components. Because they are not defined individually, data which repairs an inconsistent version also generates data updated data as the previous data was detected as corrupt and in need of update." Appellant respectfully submits that Examiner's rationale for maintaining the rejection is in error.

As an initial matter, Appellant respectfully submits that independent claim 39 does not recite "differential updates" as asserted by Examiner, but in fact, independent claim 39 recites "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between sending "clean" or "modified" data, as discussed in Levi and "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Furthermore, even if Levi's "clean" or "modified" data could be analogized to the claimed "instructions" (a point that Appellant does not concede), Appellant disagrees with Examiner's assertion that the claimed language "does not necessarily individualize these components." Actually, the claimed language of claim 39 explicitly recites that the

claimed "dedicated differential update instructions" comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version," which are clearly two types of differential update instructions.

Thus, for at least these reasons, independent claim 39 are patentable over Levi and Gary, taken alone or in any permissible combination. Appellant respectfully requests that the rejection be reversed.

E. Independent Claim 40

For example, Levi and Gary, taken alone or in any permissible combination, fail to disclose, teach, or even suggest "wherein the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version," as recited in independent claim 40.

The aforementioned element of independent claim 40 explicitly recites that the "dedicated differential update instructions" comprise two types of "differential update instructions": (1) "differential update instructions used to generate the updated data version" and (2) "differential update instructions used to repair the data that is inconsistent with the first data version." In support of the rejection, Examiner cites various passages of Levi (col. 2, lines 38-59; col. 3, lines 43-55; col. 4, lines 24-61; and col. 12, lines 47-63) as allegedly disclosing the aforementioned claim element. Appellant will address each cited passage of Levi in turn:

Col. 2, lines 38-59 of Levi discusses:

> *An aspect of some preferred embodiments of the invention relate to data redress by an output monitor. In a preferred embodiment of the invention, a copy of some or all of the data which can be transmitted is stored at a secure location. When data is proscribed from being transmitted, for example for reason of it being tampered, the output monitor obtains a "clean" copy of the data from the secure location and transmits the clean data instead. In some cases the clean data may be more limited than the original data, for example a message which indicates that data is not being transmitted. Alternatively, proscribed data is not transmitted, so that transmitted WWW pages contain blank areas. Alternatively, a standard message is transmitted, to fill in the blank areas. Alternatively or additionally, the transmitted WWW page is modified so that the page appears not to be missing data and/or so that the distortion of the page is minimized. Alternatively, the altered data is allowed to go out, with an additional message, for example, to warn the user of possible corruption. An example of such a message is a disclaimer of warranty*

for the content of the data. Another example of a message is a warning that the data may be incorrect.

In other words, this particular passage of Levi merely discusses "data redress by an output monitor." According to the passage of Levi, a substitute "clean copy" is sent, a "modified" copy is sent, or the "altered data" is sent, along with a notice that the data has been altered. However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to receiving different types of instructions, much less receiving "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 3, lines 45-55 of Levi discusses:

Alternatively or additionally, transmitting said data comprises not transmitting said data if said quality is not assured. Alternatively or additionally, said transmitting said data comprises transmitting said data if said data does not require quality assurance. Alternatively or additionally, the method comprises redressing said data if said quality assurance fails. Preferably, redressing comprises replacing said data with verified data. Preferably, said verified data is a copy of the data which was to be obtained by said data provider. Alternatively said verified data is not up-to-date copy of the data which was to be obtained by said data provider.

In other words, this passage of Levi merely discusses different ways of "transmitting data," "redressing said data," and replaying data with "verified data." However, nothing in the cited passage discloses, teaches, or even suggests anything that can be fairly analogized to different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Col. 4, lines 24-61 of Levi discusses:

In a preferred embodiment of the invention, verifying said first stage data comprises verifying a signature on a program used for said first manufacturing step. Alternatively or additionally, said first stage signing is performed by a program which performs said first manufacturing step. Alternatively or additionally, said first manufacturing step comprises database querying. Alternatively or additionally, said first manufacturing step comprises retrieving data from a remote source. Alternatively or additionally, said first and said second manufacturing steps are performed at a computing site. Alternatively or additionally, the method comprises redressing said data if said verification fails. Preferably, said redressing comprises performing a backup data manufacturing process.

There is also provided in accordance with a preferred embodiment of the invention, a method of data corruption recovery, comprising:

detecting that data to be transmitted is corrupted, after said data is prepared for transmission and while transmitting said data;

redressing said data; and

transmitting said redressed data instead of said corrupted data. Preferably, said redressing comprises retrieving replacement data from a secured location. Preferably, said secured location contains a copy of said corrupted data. Alternatively or additionally, said secured location contains a previous version of said corrupted data. Alternatively or additionally, said secured location contains a less up-to-date copy of said corrupted data.

In a preferred embodiment of the invention, said redressing comprises retrieving replacement data from a remote location. Alternatively or additionally, said redressing comprises modifying a data transmission to not include a reference to said corrupted data. Alternatively or additionally, said redressing comprises manufacturing replacement data for said corrupted data. Alternatively or additionally, said redressing is transparent to a receiver of said data transmission.

At most, the cited passage merely describes the transmission of data to correct corrupted data. However, nothing in the cited passage discloses, teaches or even suggests anything about the transmission or receipt of different types of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between the transmission of data to correct corrupted data and the transmission of instructions "used to generate the updated data version" and instructions "used to repair the data that is inconsistent with the first data version."

Col. 12, lines 47-63 of Levi discusses:

In a preferred embodiment of the invention, the backup data is a copy of the original data, maintained at a secure location which is accessible only, or mainly, by the output monitor. Thus, there is a greater probability that the backup data is not tainted. Possibly, the backup data is encrypted or stamped with a digital signature. Possibly, a copy of all the data is maintained. In one example, only data which is susceptible to corruption is maintained in copy. In another example, only data which must be available is maintained as a copy. Possibly, if data corruption is detected, a copy of the data is used to replace the site database, possibly automatically.

At most, the cited passage merely discusses aspects of Levi's backup data. Nothing in the cited passage discloses, teaches or even suggests anything about different type of instructions, much less "dedicated differential update instructions" that further include "differential update instructions used to generate the updated data version" and

"differential update instructions used to repair the data that is inconsistent with the first data version."

In the "Response to Arguments" section of the Final Office Action, Examiner asserts that "because there is no differentiating language separating the updating data version and what is effectively the repaired first version, the examine (*sic*) does not believe the amendments presented fully overcome the cited prior art." Applicant respectfully disagrees. In fact, the claims explicitly distinguish between "a first data version" and "an updated data version." *See*, claim 40 ("updating an image of stored data in a flash memory of a mobile terminal from a first data version to an updated data version"). Thus, something that is "effectively the repaired first version" is expressly distinguished from "an updated data version" since the "first version" (repaired or not) is updated to "an updated data version," according to the present claims.

Examiner issued an Advisory Action dated May 12, 2011 ("Advisory Action") that maintains the rejection. Examiner states that "Applicant argues Levi fails to disclose the two components of 'differential updates' one used to 'generate the updated data version' and one used to 'repair data that is inconsistent' however the claimed language does not necessarily individualize these components. Because they are not defined individually, data which repairs an inconsistent version also generates data updated data as the previous data was detected as corrupt and in need of update." Appellant respectfully submits that Examiner's rationale for maintaining the rejection is in error.

As an initial matter, Appellant respectfully submits that independent claim 40 does not recite "differential updates" as asserted by Examiner, but in fact, independent claim 40 recites "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version." There is a marked difference between sending "clean" or "modified" data, as discussed in Levi and "dedicated differential update instructions" which comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version."

Furthermore, even if Levi's "clean" or "modified" data could be analogized to the claimed "instructions" (a point that Appellant does not concede), Appellant disagrees

with Examiner's assertion that the claimed language "does not necessarily individualize these components." Actually, the claimed language of claim 40 explicitly recites that the claimed "dedicated differential update instructions" comprises "differential update instructions used to generate the updated data version" and "differential update instructions used to repair the data that is inconsistent with the first data version," which are clearly two types of differential update instructions.

Thus, for at least these reasons, independent claim 40 are patentable over Levi and Gary, taken alone or in any permissible combination. Appellant respectfully requests that the rejection be reversed.

CONCLUSION

The claims currently pending in the application are patentable over Levi and Gary, and the Applicants request that the Examiner's rejection thereof be reversed and the application be remanded for further prosecution.

Respectfully submitted,

/ Ronald S. Liu /

Ronald S. Liu
Registration No. 64,170
Ericsson Patent Counsel

Date: August 25, 2011

Ericsson Inc.
6300 Legacy Drive, M/S EVR1 C-11
Plano, Texas 75024

(972) 583-8512
ronald.liu@ericsson.com

CLAIMS APPENDIX

1. - 21. (Canceled)

22. (Previously Presented) A method of differentially updating an image of stored data in a mobile terminal from a first data version to an updated data version, the method comprising the steps of:

detecting whether the image of stored data in a flash memory of the mobile terminal includes one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version;

receiving dedicated differential update instructions, wherein

the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and

the dedicated differential update instructions are generated in response to detecting the image of stored data in the flash memory of the mobile terminal includes one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and

loading the dedicated differential update instructions into the flash memory of the mobile terminal;

repairing, when generating the updated data version, any such detected corrupted memory block; wherein the image of stored data in the flash memory is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.

23. (Previously Presented) The method according to claim 22, further comprising generating the differential update instructions based on information about detected corrupted memory blocks, if any.

24. (Previously Presented) The method according to claim 23, wherein the differential update instructions include update data and the step of generating the updated data version further comprises the step of replacing data stored in predetermined one or more memory blocks by the update data.

25. (Previously Presented) The method according to claim 24, wherein the update data includes one or more repaired memory blocks of data consistent with the updated data version, the one or more repaired memory blocks of data corresponding to the detected one or more corrupted memory blocks of data, if any.

26. (Previously Presented) The method according to claim 23, wherein the step of generating the differential update instructions further comprises the step of generating instructions by the processor of the mobile terminal to cause the processor of the mobile terminal to generate the updated data version from the image of the stored data, excluding any detected one or more corrupted memory blocks from the differential update instructions.

27. (Previously Presented) The method according to claim 23, wherein the step of generating the differential update instructions is performed by a remote data processing system.

28. (Previously Presented) The method according to claim 27, further comprising the step of the mobile terminal and the remote data processing system communicating via a wireless communications link.

29. (Previously Presented) The method according to claim 28, further comprising the step of the mobile terminal and the remote data processing system communicating via an Internet Protocol.

30. (Previously Presented) The method according to claim 27, wherein the step of detecting is performed by the mobile terminal and the detecting further comprises the step of transmitting information about the detected one or more corrupted memory blocks from the mobile terminal to the remote data processing system.

31. (Previously Presented) The method according to claim 27, wherein the method further comprises the step of transmitting information about the image of the stored data from the mobile terminal to the remote data processing system and wherein the step of detecting is performed by the remote data processing system from the transmitted information.

32. (Previously Presented) The method according to claim 22, wherein the step of detecting further comprises the steps of:

calculating a number of checksums by the processor of the mobile terminal, wherein each checksum corresponds to a corresponding memory block of data stored in the flash memory of the mobile terminal; and

comparing the calculated checksums with a number of reference checksums to identify any corrupted memory block of data.

33. (Previously Presented) The method according to claim 32, wherein the reference checksums are stored in the flash memory of the mobile terminal and further comprising the step of performing the step of comparing by the mobile terminal.

34. (Previously Presented) The method according to claim 33, further comprising the step of integrity protecting the reference checksums stored in the mobile terminal by a message authentication code.

35. (Previously Presented) The method according to claim 32, further comprising the steps of:

storing the reference checksums on a remote data processing system wherein the transmitted information comprises the calculated checksums; and

wherein the detecting step further comprises the step of comparing the transmitted calculated checksums by the remote data processing system with the number of reference checksums stored on the remote data processing system.

36. (Previously Presented) The method according to claim 32, wherein the calculating step further comprises the step of calculating the checksums as a cryptographically strong one-way hash function of the corresponding memory block of the image of the stored data.

37. (Previously Presented) A mobile terminal comprising:

a flash memory for storing an image of data;

communications means adapted to receive from a data processing system dedicated differential update instructions for updating the image of data stored in the flash memory from a first data version to an updated data version;

processing means adapted to generate the updated data version from the image of the stored data and from the received dedicated differential update instructions, wherein the processing means is further adapted to:

generate information from the image of the stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version;

communicate the generated information via the communications means to the data processing system for generating the dedicated differential update instructions, wherein

the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and

the dedicated differential update instructions are generated in response to the data processing system receiving the information from the image of stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and

repair any such detected corrupted memory block; wherein the image of stored data in the flash memory is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.

38. (Previously Presented) A data processing system for facilitating differentially updating an image of stored data in a mobile terminal from a first data version to an updated data version, the data processing system comprising:

means for loading dedicated differential update instructions into a flash memory of the mobile terminal, the dedicated differential update instructions causing a processor of the mobile terminal to generate the updated data version from the an image of stored data and the loaded dedicated differential update instructions;

the data processing system further comprising:

means for receiving information from the mobile terminal indicative of the presence of one or more corrupted memory blocks wherein the image of stored data is inconsistent with the first data version; and

processing means adapted to generate the dedicated differential update instructions from the first and updated data versions and from received information, wherein

the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and

the dedicated differential update instructions are generated in response to receiving information from the mobile terminal indicating that the image of stored data in the flash memory includes one or more

corrupted memory blocks having stored therein data that is inconsistent with the first data version; and

wherein the differential update instructions used to repair the data that is inconsistent with the first data version are adapted to cause the processor of the mobile terminal to repair any such detected corrupted memory block; wherein the image of stored data in the flash memory of the mobile terminal is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.

39. (Previously Presented) A computer program comprising program code means embodied on a computer readable medium to be loaded into a flash memory means and executed by a processor means and adapted to cause a mobile terminal to differentially update an image of stored data in the flash memory of the mobile terminal from a first data version to an updated data version by performing the following steps, when the computer program is executed by the processor means on the mobile terminal:

generating information from the image of stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with a first data version;

loading dedicated differential update instructions into the processor of the mobile terminal, wherein

the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and

the dedicated differential update instructions are generated in response to the information from the image of stored data indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version; and

generating the updated data version by the processor of the mobile terminal from the stored data and the loaded dedicated differential update instructions, including

repairing any such detected corrupted memory block; wherein the image of stored data in the flash memory of the mobile terminal is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.

40. (Previously Presented) A computer program comprising program code means embodied on a computer readable medium to be loaded into a memory means and executed by a processor means and adapted to cause a data processing system to facilitate differentially updating an image of stored data in a flash memory of a mobile terminal from a first data version to an updated data version by performing the following steps, when the computer program is executed by the processor on the data processing system:

generating dedicated differential update instructions from the first and updated data versions and from information received from the mobile terminal, wherein the received information is indicative of the presence of one or more corrupted memory blocks having stored therein data that is inconsistent with the first data version, wherein

the dedicated differential update instructions comprise differential update instructions used to generate the updated data version and differential update instructions used to repair the data that is inconsistent with the first data version, and

the dedicated differential update instructions are generated in response to receiving the received information; and

loading the generated dedicated differential update instructions into the flash memory of the mobile terminal, the dedicated differential update instructions causing the processor of the mobile terminal to generate the updated data version from the stored data and the loaded dedicated differential update instructions, wherein the image of stored data in the flash memory of the mobile terminal is updated in-place such that data of the first data version is reused and reorganized to generate the updated data version.

41. (Previously Presented) The mobile terminal according to claim 37, in combination with a mobile communications network.

42. (Previously Presented) The method according to claim 23, wherein the step of generating the differential update instructions is performed by a processor of the mobile terminal.

* * *

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.